

# **Checkpoint III Base de Datos**

Matias Amil

Matias Fernandez

Victoria Stancic

Natalia Varesi

## PROYECTO DHTube III

Nos encontramos en el tercer punto de encuentro dentro del proyecto DhTube. La empresa nos pidió una reunión más y junto con ellos cerramos el modelado de la base de datos.

### Enunciado

La empresa DHTube nos pidió una tercera reunión. En este hito, tenemos que mostrar la base de datos que construimos e informes más completos. Según lo que relevamos, generamos un punteo de todos lo solicitado y comenzaremos a trabajar en lo que tenemos que entregar en esa reunión.

### Checkpoint III: Integrador de SQL

Realizar los siguientes reportes:

1. Generar un reporte que indique la cantidad de usuarios por país, mostrando el nombre del país y su cantidad.

```
select count(idUsuario) as Cantidad, pais.nombre from usuario
right join pais on pais.idPais = usuario.Pais_idPais
group by pais.nombre;
```

```
3 * select count(idUsuario) as Cantidad, pais.nombre from usuario
4   right join pais on pais.idPais = usuario.Pais_idPais
5   group by pais.nombre;
```

2. Mostrar el Top 5 de avatars utilizados por los usuarios, listando el nombre del avatar y la cantidad de veces utilizado.

```
select avatar.nombre, count(usuario.Avatar_idAvatar) as Cantidad from avatar
join usuario on usuario.Avatar_idAvatar = avatar.idAvatar
group by avatar.nombre
order by count(usuario.Avatar_idAvatar) desc
limit 5;
```

```
10 * select avatar.nombre, count(usuario.Avatar_idAvatar) as Cantidad from avatar
11   join usuario on usuario.Avatar_idAvatar = avatar.idAvatar
12   group by avatar.nombre
13   order by count(usuario.Avatar_idAvatar) desc
14   limit 5;
```

3. Emitir un listado de todas las playlists públicas, informando el nombre del usuario que la creó, el título de cada video que posee y el nombre y año de creación de la playlist.

```
select playlist.*, usuario.nombre as Usuario, video.titulo as Título,
playlist.nombre as Nombre_Playlist, playlist.fechaCreacion as Fecha_Creacion
from playlist
join usuario on usuario.idUsuario = playlist.Usuario_idUsuario
join playlist_video on playlist_video.Playlist_idPlaylist = playlist.idPlaylist
join video on video.idVideo = playlist_video.Video_idVideo
where playlist.privado = 0
group by idPlaylist;
```

```

19 * select playlist.*, usuario.nombre as Usuario, video.titulo as Título,
20     playlist.nombre as Nombre_Playlist, playlist.fechaCreacion as Fecha_Creacion
21     from playlist
22     join usuario on usuario.idUsuario = playlist.Usuario_idUsuario
23     join playlist_video on playlist_video.Playlist_idPlaylist = playlist.idPlaylist
24     join video on video.idVideo = playlist_video.Video_idVideo
25     where playlist.privado = 0
26     group by idPlaylist;
27

```

4. Listar las 10 etiquetas menos usadas.

```

select etiqueta.nombre as Nombre_Etiqueta, count(etiqueta.idEtiqueta) as Veces_Usada
from etiqueta
join video_etiqueta on video_etiqueta.Etiqueta_idEtiqueta = etiqueta.idEtiqueta
group by nombre_etiqueta
order by veces_usada limit 10;

```

```

29 * select etiqueta.nombre as Nombre_Etiqueta, count(etiqueta.idEtiqueta) as Veces_Usada from etiqueta
30     join video_etiqueta on video_etiqueta.Etiqueta_idEtiqueta = etiqueta.idEtiqueta
31     group by nombre_etiqueta
32     order by veces_usada limit 10;
33

```

5. Generar un reporte de las últimas 10 reacciones, listando la fecha, el nombre de la reacción, el id y nombre de usuario, y el título del video.

```

select tiporeaccion.nombre as Nombre_reaccion, reaccion.fecha, usuario.idUsuario as
UsuarioID,
    usuario.nombre Nombre_Usuario, video.titulo Titulo_Video from reaccion
join tiporeaccion on tiporeaccion.idTipoReaccion =
reaccion.TipoReaccion_idTipoReaccion
join usuario on usuario.idUsuario = reaccion.Usuario_idUsuario
join video on video.idVideo = reaccion.Video_idVideo
order by reaccion.fecha limit 10;

```

```

34 -- 5. Generar un reporte de las últimas 10 reacciones, listando la fecha, el
35 -- nombre de la reacción, el id y nombre de usuario, y el título del video.
36 * select tiporeaccion.nombre as Nombre_reaccion, reaccion.fecha, usuario.idUsuario as UsuarioID,
37     usuario.nombre Nombre_Usuario, video.titulo Titulo_Video from reaccion
38     join tiporeaccion on tiporeaccion.idTipoReaccion = reaccion.TipoReaccion_idTipoReaccion
39     join usuario on usuario.idUsuario = reaccion.Usuario_idUsuario
40     join video on video.idVideo = reaccion.Video_idVideo
41     order by reaccion.fecha limit 10;
42

```

6. Mostrar por usuario la cantidad de playlists que posee, pero solo para aquellos que tengan más de 1 playlist creada en el año 2021.

```

select usuario.nombre, count(idPlaylist)
FROM usuario
JOIN playlist
ON playlist.Usuario_idUsuario = usuario.idUsuario

```

where playlist.fechaCreacion BETWEEN "2021-00-00" and "2021-12-31"  
group by usuario.nombre;

- ```
select usuario.nombre, count(idPlaylist)
FROM usuario
JOIN playlist
ON playlist.Usuario_idUsuario = usuario.idUsuario
where playlist.fechaCreacion BETWEEN "2021-00-00" and "2021-12-31"
group by usuario.nombre;
```

7. Generar un reporte de las reacciones generadas en el 2021, con el siguiente formato :

Nombre de Reacción Nombre de Usuario Título de Video Fecha

```
SELECT idReaccion "Nombre de Reacción", Usuario_idUsuario "Nombre de Usuario",
Video_idVideo "Título de Video", fecha "Fecha"
FROM reaccion;
```

- ```
SELECT idReaccion "Nombre de Reacción", Usuario_idUsuario "Nombre de Usuario",
Video_idVideo "Título de Video", fecha "Fecha"
FROM reaccion;
```

8. Listar la cantidad de videos según sean públicos o privados.

```
SELECT count(*), privado
FROM video
GROUP BY privado;
```

- ```
SELECT count(*), privado
FROM video
GROUP BY privado;
```

9. Generar un reporte con los nombres de los usuario que no poseen ninguna Playlist.

```
SELECT usuario.nombre
FROM usuario
LEFT JOIN playlist
ON usuario.idUsuario = playlist.Usuario_idUsuario
WHERE playlist.idPlaylist is null;
```

- ```
SELECT usuario.nombre
FROM usuario
LEFT JOIN playlist
ON usuario.idUsuario = playlist.Usuario_idUsuario
WHERE playlist.idPlaylist is null;
```

10. Listar todos los videos hayan o no recibido reacciones en el último mes; indicar cuántas reacciones tuvieron y si han sido reproducidos o no. El listado debe estar ordenado alfabéticamente por título del video.

```
SELECT video.titulo, count(video.cantidadReproducciones) AS cantidadReproducciones,
count(reaccion.idReaccion) AS cantidadReacciones, date(reaccion.fecha) as fecha
FROM video
LEFT JOIN reaccion
ON video.idVideo = reaccion.video_idvideo
where reaccion.fecha between "2021-05-00" and "2021-05-31"
group by video.titulo
ORDER BY reaccion.fecha desc;
```

- ```
SELECT video.titulo, count(video.cantidadReproducciones) AS cantidadReproducciones,
count(reaccion.idReaccion) AS cantidadReacciones, date(reaccion.fecha) as fecha
FROM video
LEFT JOIN reaccion
ON video.idVideo = reaccion.video_idvideo
where reaccion.fecha between "2021-05-00" and "2021-05-31"
group by video.titulo
ORDER BY reaccion.fecha desc;
```

11. Generar un reporte con el nombre del usuario y el título de videos del usuario que no pertenecen a ninguna playlist.

```
select usuario.nombre, video.titulo as Titulo_video, playlist.idPlaylist from
usuario
left join video on usuario.idUsuario = video.Usuario_idUsuario
left join playlist_video on video.idVideo = playlist_video.Video_idVideo
left join playlist on playlist_video.Playlist_idPlaylist
where playlist.idPlaylist is null;
```

```
72 * select usuario.nombre, video.titulo as Titulo_video, playlist.idPlaylist from usuario
73 left join video on usuario.idUsuario = video.Usuario_idUsuario
74 left join playlist_video on video.idVideo = playlist_video.Video_idVideo
75 left join playlist on playlist_video.Playlist_idPlaylist
76 where playlist.idPlaylist is null;
```

12. Listar a todos los usuarios que no poseen ningún video.

```
select usuario.idUsuario, usuario.nombre, video.idVideo from usuario
left join video
```

```
on usuario.idUsuario = video.Usuario_idUsuario
where Video.idVideo is null;
```

```
7 • select usuario.idUsuario, usuario.nombre, video.idVideo from usuario
8 left join video
9 on usuario.idUsuario = video.Usuario_idUsuario
10 where Video.idVideo is null;
```

13. Listar la cantidad total de reacciones por cada tipo de reacción, en el periodo del 01-01-2021 al 01-04-2021

```
SELECT sum(idreaccion) cantidadTotal, tiporeaccion.nombre
FROM reaccion
join tiporeaccion
ON reaccion.TipoReaccion_idTipoReaccion = tiporeaccion.idTipoReaccion
WHERE reaccion.fecha BETWEEN "2021-01-01" and "2021-04-21"
group by tiporeaccion.nombre;
```

```
• SELECT sum(idreaccion) cantidadTotal, tiporeaccion.nombre
FROM reaccion
join tiporeaccion
ON reaccion.TipoReaccion_idTipoReaccion = tiporeaccion.idTipoReaccion
WHERE reaccion.fecha BETWEEN "2021-01-01" and "2021-04-21"
group by tiporeaccion.nombre;
```

14. Listar los videos que tienen los usuarios cuyo nombre contiene la letra "a" y son del país Argentina.

15. Generar un informe estilo ranking de avatars utilizados por país.

```
select count(avatar.idAvatar) as Ranking, pais.nombre, avatar.nombre as avatar from
usuario
join avatar on usuario.Avatar_idAvatar = avatar.idAvatar
join pais on pais.idPais = usuario.Pais_idPais
group by pais.idPais, avatar.idAvatar;
```

```
93 -- 15. Generar un informe estilo ranking de avatars utilizados por país.
94 • select count(avatar.idAvatar) as Ranking, pais.nombre, avatar.nombre as avatar from usuario
95 join avatar on usuario.Avatar_idAvatar = avatar.idAvatar
96 join pais on pais.idPais = usuario.Pais_idPais
97 group by pais.idPais, avatar.idAvatar;
98
```

16. Generar un reporte de todos los videos, mostrando los que poseen reacciones y cuántas veces han sido reproducidos.

17. Mostrar los 5 videos de menor duración, listando el título del video y el nombre de usuario y país al que corresponde.

```
select video.duracion, titulo, min(duracion)
from video
join usuario on video.Usuario_idUsuario = usuario.idUsuario
```

```
order by video.duracion limit 5;
```

```
select video.duracion, titulo, min(duracion)
from video
join usuario on video.Usuario_idUsuario = usuario.idUsuario
order by video.duracion limit 5;
```

18. Listar el usuario brasilero con más reacciones durante el 2021.

19. Generar un reporte listando los usuarios, sus canales, playlists y los videos que integran esas playlists.

```
select * from usuario
join canal on canal.Usuario_idUsuario = usuario.idUsuario
join playlist on playlist.Usuario_idUsuario = usuario.idUsuario
join playlist_video on playlist_video.Playlist_idPlaylist = playlist.idPlaylist
join video on playlist_video.Video_idVideo = video.idVideo;
group by usuario
```

20. Listar todas las playlists que posean menos de 3 videos, mostrando el nombre de usuario y el avatar que posee.

```
select count(playlist_video.Video_idVideo) as Cantidad_videos, playlist.nombre,
avatar.nombre from playlist
join usuario on usuario.idUsuario = playlist.Usuario_idUsuario
join avatar on avatar.idAvatar = usuario.Avatar_idAvatar
join playlist_video on playlist.idPlaylist = playlist_video.Playlist_idPlaylist
join video on playlist_video.Video_idVideo = video.idVideo
group by playlist.idPlaylist
having count(playlist_video.Video_idVideo) < 3;
```

```
118 * select count(playlist_video.Video_idVideo) as Cantidad_videos, playlist.nombre, avatar.nombre from playlist
119 join usuario on usuario.idUsuario = playlist.Usuario_idUsuario
120 join avatar on avatar.idAvatar = usuario.Avatar_idAvatar
121 join playlist_video on playlist.idPlaylist = playlist_video.Playlist_idPlaylist
122 join video on playlist_video.Video_idVideo = video.idVideo
123 group by playlist.idPlaylist
124 having count(playlist_video.Video_idVideo) < 3;
125
```